BECKER • KURIG • STRAUS
MÜNCHEN – BERLIN

Patentanwälte Becker Kurig Straus • Bavariastr. 7 • D-80336 München

Patent- och registreringsverket
Box 5055
10242 Stockholm
Schweden

Via telefax in advance (+46-8667-7288)
Original by registered mail

Galileo®

Patent- und Rechtsanwälte
European Patent Attorneys
European Trademark Attorneys

Dr. Eberhard Becker, Chem.
Dr. Thomas Kurig, Dipl.-Phys.
Dr. Alexander Straus, Dipl.-Chem.
Dr. Roman Vuille[1], Dipl.-Chem.
Friedrich von Braun, Rechtsanwalt

Peter Kylin, MSc.[2]
Magnus Hynell, MSc.[2]
Annika Björkman, MSc.[2]
Ivar Andréasson, MSc.[2]
Eva Lena Jansson[2]
Magnus Aspeby, MSc.[2] of counsel
Lars E. Johansson, MSc.[2] of counsel

Bavariastrasse 7
D-80336 München
Tel.: +49-89-746 303 0
Fax: +49-89-746 303 11
info@galileolaw.de
www.galileolaw.de

December 23, 2004

PCT-Application PCT/IB02/04990
Applicant / Owner: Nokia Corporation
Title: Method and device for determining and outputting the ...
Our Ref.: 51242 WO (KG/BK)

In Response to the Official Communication Following Rule 66, dated September 30, 2004:

I. New Documents

We submit herewith a set of new claims 1 to 15.

The new independent claims 1 and 14 have been restricted to a pairwise string comparison method using allocated position labels to each of said entities in the string, wherein same entities are numbered according to their relative position in accordance with the position label determining similar data entities with the same order in said second data string.

The new independent claims 1 is disclosed in the original claims 1, 2 and 4.

The new dependent claims 2 and 3 are disclosed in original claims 3 and 5.

The new dependent claims 4 to 13 are disclosed in original claims 6 to 15

The new independent claim 14 is disclosed in the original claim 16 and the method claims 1, 2 and 4. The new claim 14 has been adapted to the wording of the new method claim 1.

The new dependent claim 15 is disclosed in original claim 17.

The original claims 2 and 4 have been deleted.

II. Object of the present invention

It is the object of the present invention to improve the present pattern recognition techniques by increasing the robustness of existing string and sequence comparison techniques and to reduce the search time of an algorithm as compared to present similarity determination algorithms. (See page 3, lines 4-8.)

III. The invention as claimed

The present invention as claimed pertains to a method for determining and outputting a similarity measure between two data strings as claimed in claim 1, and to a device for executing said method as claimed in claim 14.

The method comprising receiving a first and a second data string. The method proceeds with determining pairs of consecutively following data entities in said first data string and determining the relative positions of said pairs of consecutively following data entities in said first data string. That is, the strings are "chopped" into small pieces comprising each a pair of consecutively following data entities. Then position labels are allocated to each of said entities in the string. Then same entities are numbered according to their relative position in accordance with the position label determining similar data entities with the same order in said second data string. (I.e. same types pieces are each numbered consecutively). The second string is examined for the relative positions of these determined data entities.

The method of the present invention is based on a two-note value comparison algorithm using a relative position data of two successive note values to search for the respective positions (maybe not succeeding) of these two notes in the corresponding sequences in the

second string.

Then a matching measure is determined by determining how far the relative positions of data entities in said second data string match with the relative positions of consecutively following data entities in said first data string. Finally a similarity measure is outputted which corresponds to the matching measure of at least one comparison result.

The present approach is based on the idea that music is not only based on the strict string of notes but may better be described by the "first derivative" of the notes, and on the distribution of these derivatives and the distance of two notes of the respective "derivative" in a target string of notes.

The present invention could allow recognizing e.g. transcripted pieces of music if e.g. the notation of note pairs may be based on a standardized first tone and a normalized difference.

VI. State of the Art

The examining division has cited only two documents representing the state of the art.

D1: CHENG YANG: "MACS: Music Audio Characteristics Sequence Indexing for Similarity Retrieval". In IEEE Workshop on Applications of Signal Processing to Audio and Acoustics. 2001. 21-24 Oct. New York.

D2: US 5402339 A

D1 discloses a method for matching audio data. Audio data is firstly converted into a continuous spectra from which a string of elements is derived. That is the strings in D1 are not time-normalized. In order to compare two strings, the method comprises indexing means in order to capture the relative order of the elements included in the string. A matching procedure is then performed; each match contains a tuple (query-offset, matching-offset). A "good" match occurs when the relative order of the elements in the query string and the reference string agrees.

However, D1 is based on "tuples" composed of elements of the two strings. These tuples are then projected on such "tuples" on a two-dimension plane to determine a correlation

value to determine a matching value. To reduce the processing expenditure a preprocessing step is used to simplify the data processing to determine a matching factor.

D2 discloses an apparatus for retrieving musical information. A music piece is converted to a string of elements, where each element represents note data, each characterized by a tone height and a duration of the tone. The apparatus also include means for producing position data representing positions at which place a note is positioned within the music information. This position data is also used as an index of the position of the music information by storing said position data, indicating the relationship between note data items and position data. The absolute order of the note data is considered when matching strings of musical information. D2 seems not to disclose any kind of relative note position based similarity algorithm. Additionally, D2 relay so a histogram-based approach for providing a similarity measure for two strings of music data (see description of Figure 1B, col. 3, l. 65 to col. 4, l. 9).

Additionally D2 uses a "creep algorithm" to shift both sequences against each other if the absolute position data can not be matched (see col. 6, l. 23- 27). This creeping of (usually the smaller) string along the longer string is required to eliminate a certain offset between the strings. In D2 this creeping process is improved by determining the least frequent note data item and using this note data item for finding the next checking position for matching the strings.

V. Novelty

The examiner has not objected to the novelty of the original claims 3, 5-11, 14 and 17. The new independent claims 1 and 14 are based on the features of the original claims 1, 2, 5 and 16, 2, 5 respectively. Therefore it is expected that the new claims are considered to be novel by the examining division.

D1 does not disclose pairs of notes for determining a similarity measure. D1 does not disclose the numbering same entities according to their relative position in the search string. Thus, the new claims are novel with respect to D1.

D2 does not disclose the use of relative position information for determining a similarity measure. D2 does only disclose an absolute indexing but no consecutive numbering of same entities according to their relative position in the search string, thus the new claims are novel with respect to D2.

VI. Inventive step

The examiner objects to an inventive step of all pending claims with regard to the cited documents D1 and D2.

The examiner has not cited one of the both documents to be the closest state of the art. It is expected that D2 represents the closest state of the art as D2 and the present invention are based on more or less "discrete" or integer note or music data values, using a match algorithm between discrete pairs and not preprocessed filtered intensity graphs, such as D1. D2 and the present invention use tempo normalized algorithms.

The disclosure of D2 is based on a comparison method for determining if a short passage is present in a long music data string by using a comparison method. In D1 the method is based on a strict one-to-one representation, allowing no mistakes and or deviation from the original search sequence. In claim 12 the note data items are related to an absolute position of note data items for finding a certain passage fast by detecting a first note data item followed by a determination if this note data is followed in both strings by the same note data item. That is, the D2 suggests to use a determine for each note data item in the search if it is followed by a respective note data item in the target string with reference to an absolute position index in the search string.

As the method of D2 is solely based on the use of an index and additionally allows the use of different offsets between the search string and the target string. D2 relies in a strict translational identity measure. The restricted number of possible pitches increases the number of possible start positions for the comparison. Additionally, D2 proposes to select the "rarest" note data item for speeding up the determining of a certain threshold if the comparing algorithm. D2 also suggests to speed up the offset process by introducing a larger number of possible "pitches" or note data items by using two-tone note data items, and increase the number of possible values in for a note data item, which increased the step width of a offset determination.

D2 as such is not capable of determining only a similarity in case e.g. a single note data element in the middle of the search string is missing, as this measure disturbs the overall similarity caused by the absolute indexing.

To speed up the search for the right search string D2 suggests to use histogram data to determine in a fast process if the search string may be present in a certain target string. That is D2 suggests plausibility check before selecting a string for a search.

Anyhow, D2 is still only based on a strict one-to-one relationship between the search string and the target string, as in any case D2 relies on the same data format of the note data items both as notes or both as pairs of notes.

An artisan finds no suggestion to use different data formats in the search and the target strings. D2 as such uses a tempo normalized algorithm relying on minim and crotchet note duration values. That is, the normalized note data structure in D2 in connection with a strict single pitch note string removes any chance of missing tone or any irregularity in the absolute positioning approach of D2. In case of D2 it may not happen that as single missing note data item can disturb the absolute numbering or indexing of the search string.

Following the principles of D2 an artisan would select higher tuples e.g. three note value based note data items to increase the meaningfulness of histogram data. D2 would also suggest to select higher tuples e.g. three note value based note data items to improve the threshold determination for the comparison algorithm. Finally an artisan following the principles of D2 would end up at a complete data base of all strings comprised in the target strings to allow a "single access" to determine instantaneously all identical sub-strings of all target strings in the database.

That is D2 alone discloses no idea for dealing with damaged or multilated search strings. D2 alone shows no suggestion for comparing different note data items (consecutive pairs in the first string and the respective (n9on-consecutive) pair of note data items in second string). The fact that D2 relates on a normalized search string economizes the necessity for any kind of fault resistance in the search algorithm. D2 only relates to an absolute indexing of the single items in the strings, as provide no suggestion for a relative numbering of similar note data items.

All these features are neither addressed nor indicated in D2, thus it is not clear why an artisan would have been caused to introduce these features in the method of D2. This is especially not necessary as the source for a search string is always a "trusted source" such as in D2 musical information representing a music piece is supplied from an external unit (e.g. a computer system, an electrical musical instrument, a music sampler or a reproducing device). That is, there is no need to expect a faulty string of note data items as

a starting point for a music search. Thus, in D2 alone there is no need for any kind of fault resistance or any kind of non-absolute indexing in the search string. This is especially true, as starting from "faultless" sequences any fault resistant algorithm would only increase the processing costs. D2 is not directed to find any similarity, but only identical matches.

When confronted with a problem of similarity, an artisan may use any kind of similarity evaluation on the basis of an absolute indexing scheme, but finds no indication to deviate from a one-to-one comparison scheme.

Thus, it is shown that the subject matter of the present claims as defined in the independent claims 1 and 14 is to be regarded as being inventive over D2 alone.

D2 as such is restricted to single tone sequences as each of note data items being a set of values of a period of a single sound denoted by a musical note and a scale level of the single sound denoted by the musical note. That is, D2 is not capable of dealing with two-part pieces of music. Additionally, D2 is not capable of dealing with pieces of music allowing to play different note values simultaneously such as e.g. a piano.

The algorithm disclosed in D1 is however based on multidimensional vectors allowing many different tones in a search and target piece of music. The whole approach in D2 considerably differs from the approach of D2. D1 uses comb filters, power plots and e.g. 24 dimensional vectors for representing a piece of music. Compared to the slightly simple approach of the present invention and of D2 there are two different requirements for processing power and storage capacity that it seems unlikely that an artisan would combine any features of these documents. Especially D1 notes on col. 2, l. 6-7 "Because MIDI-style music is very structured string matching or text searching methods can be applied. This sentence shows that D1 is directed to comparison method for raw audio data. This is the only reason why the tempo-matching algorithm of D1 figure 5 is required. In the field of structured note data items such an algorithm would be not necessarry.

The different approaches of D1 (raw music data) and D2 (MIDI files) would keep an artisan from combining the disclosures D1 and D2. As an artisan would refrain from any combination of D1 and D2 the new claims are to be regarded as being inventive with respect to the disclosure of the cited documents D1 and D2.

In view of the fact that both documents do not disclose any numbering of same entities according to their relative position in the search string (as this is not necessary as D2

already uses an absolute position indexing, and D1 relies especially on the relative absolute distribution to achieve a matching measure,) this feature may not be provided by any combination of D1 and D2. As this feature is not disclosed in D1 or D2 the present invention as disclosed in the new claims is to be regarded as being inventive with respect to the disclosure of the cited documents D1 and D2.

VIII. Requests

In view of the above arguments it is assumed that the Examiner's objections have been overcome, and it is therefore respectfully requested that the claims 1 to 15 as presently on file are acknowledged as being new and inventive. Therefore, issuance of a favorable IPER is kindly requested.

Dr. Thomas Kurig
(Patent Attorney)

Enclosure
Set of new claims 1 to 15 (+ track changes version)

International Application No. PCT/IB02/04990
Applicant: Nokia Corporation
Date December 20, 2004

Claims

1. Method for determining and outputting a similarity measure between two data strings, each data string comprising data entities, comprising:
- receiving a first data string,
- receiving a second data string,

characterized by

- determining pairs of consecutively following data entities in said first data string,
- determining the relative positions of said pairs of consecutively following data entities in said first data string,
- allocating a position label to each of said data entities in the first data string,
- numbering same data entities according to their relative position in accordance with the position label,
- determining similar data entities with the same order in said second data string,
- determining the relative positions of said determined data entities in said second data string,
- determining a matching measure by determining how far the relative positions of data entities in said second data string match with the relative positions of consecutively following data entities in said first -data string, and
- outputting a similarity measure which corresponds to the matching measure of at least one comparison result.

~~2.Method according to claim 1, wherein pairs of consecutively following data entities are determined in said first data string.~~

~~3.~~2. Method according to claim 1, further comprising:
- determining at least one error limit for at least one of said entities,
- considering said at least one error limit during said determination of said matching measure.

~~4.Method according to claim 2, further comprising, allocating a position label to each of said entities in the string, and numbering same entities according to their relative position in accordance with the position label.~~

~~5.~~3. Method according to claim ~~2~~1, further comprising:

- determining a first distance between said two data entities of consecutively following data entities in said first data string,
- determining a second distance of said two data entities determined in said second data string,
- determine a difference between said first and second distances, and
- considering said difference during said determination of said matching measure.

~~6.~~4. Method according to claim 1, further comprising:
- storing said second string together with said similarity measure.

~~7.~~5. Method according to claim 1, further comprising:
- determining a threshold for said similarity measure, and
- outputting said second string, if said determined similarity measure at least equals said threshold.

~~8.~~6. Method according to claim ~~7~~5, further comprising:
- repeating said determination of said similarity measure with a number of second strings, and
- determining said threshold in correspondence with a number of second strings to be outputted.

~~9.~~7. Method according to claim 1, further comprising:
- analyzing the first string for entities not present in the first string, and
- suppressing in the second string all said entities not present in said first string.

~~10.~~8. Method according to claim ~~9~~7, further comprising:
- determining the number of entities that are present in the second string, but are not present in the first string, as a second similarity measure.

~~11.~~9. Method according to claim ~~10~~8, further comprising:
- determining a section within said second string comprising at least the same number of entities that are simultaneously present in both strings.

~~12.~~10. Software tool comprising program code means stored on a computer readable medium for carrying out the method of anyone of claims 1 to ~~12~~ 9 when said software tool is run on a computer or network device.

~~13.~~11. Computer program product comprising program code means stored on a computer readable medium for carrying out the method of anyone of claims 1 to ~~12~~ 9 when said

program product is run on a computer or network device.

~~14.~~12. Computer program product comprising program code, downloadable from a server for carrying out the method of anyone of claims 1 to ~~12~~ 9 when said program product is run on a computer or network device.

~~15.~~13. Computer data signal embodied in a carrier wave and representing a program that instructs a computer to perform the steps of the method of anyone of claims 1 to ~~12~~9.

~~16.~~14. Electronic device for determining and outputting a similarity measure between two data strings each comprising data entities, comprising:

- a component for receiving a first data string of entities and a second data string of entities,
- a processing unit being connected to said receiving component, said processing unit being configured to determine ~~at~~ pairs ~~least one tuple~~ of consecutively following data entities in said first data string, said processing unit being configured to determine the relative positions of said pairs ~~at least one tuple~~ of consecutively following data entities in said first data string, and for allocating a position label to each of said data entities in the first data string, and numbering same data entities according to their relative position in accordance with the position label; said processing unit being configured to determine similar data entities with the same order in said second data string, and to determine the relative positions of said determined ~~at least one tuple of similar consecutively following~~ data entities in said second data string, , , said processing unit being configured to determine a matching measure by determining~~comparing~~ how far the relative positions of data entities~~the at least one tuple of similar consecutively following data entities~~ in said second~~first~~ data string matches with the relative positions of ~~said at least one tuple of similar~~ consecutively following data entities in said first~~second~~ data string, and said processing unit being configured to output a similarity measure which corresponds to the matching measure of at least one comparison result, and
- an interface being connected to said for processing unit for outputting said similarity measure.

~~17.~~15. Electronic device according to claim ~~17~~14, further comprising a storage connected to said processing unit for storing received strings and said determined similarity measures.

International Application No. PCT/IB02/04990
Applicant: Nokia Corporation
Date December 20, 2004

Claims

1. Method for determining and outputting a similarity measure between two data strings, each data string comprising data entities, comprising:
- receiving a first data string,
- receiving a second data string,
  characterized by
- determining pairs of consecutively following data entities in said first data string,
- determining the relative positions of said pairs of consecutively following data entities in said first data string,
- allocating a position label to each of said data entities in the first data string,
- numbering same data entities according to their relative position in accordance with the position label,
- determining similar data entities with the same order in said second data string,
- determining the relative positions of said determined data entities in said second data string,
- determining a matching measure by determining how far the relative positions of data entities in said second data string match with the relative positions of consecutively following data entities in said first data string, and
- outputting a similarity measure which corresponds to the matching measure of at least one comparison result.

2. Method according to claim 1, further comprising:
- determining at least one error limit for at least one of said entities,
- considering said at least one error limit during said determination of said matching measure.

3. Method according to claim 1, further comprising:
- determining a first distance between said two data entities of consecutively following data entities in said first data string,
- determining a second distance of said two data entities determined in said second data string,
- determine a difference between said first and second distances, and
- considering said difference during said determination of said matching measure.

4. Method according to claim 1, further comprising:
- storing said second string together with said similarity measure.

5. Method according to claim 1, further comprising:
- determining a threshold for said similarity measure, and
- outputting said second string, if said determined similarity measure at least equals said threshold.

6. Method according to claim 5, further comprising:
- repeating said determination of said similarity measure with a number of second strings, and
- determining said threshold in correspondence with a number of second strings to be outputted.

7. Method according to claim 1, further comprising:
- analyzing the first string for entities not present in the first string, and
- suppressing in the second string all said entities not present in said first string.

8. Method according to claim 7, further comprising:
- determining the number of entities that are present in the second string, but are not present in the first string, as a second similarity measure.

9. Method according to claim 8, further comprising:
- determining a section within said second string comprising at least the same number of entities that are simultaneously present in both strings.

10. Software tool comprising program code means stored on a computer readable medium for carrying out the method of anyone of claims 1 to 9 when said software tool is run on a computer or network device.

11. Computer program product comprising program code means stored on a computer readable medium for carrying out the method of anyone of claims 1 to 9 when said program product is run on a computer or network device.

12. Computer program product comprising program code, downloadable from a server for carrying out the method of anyone of claims 1 to 9 when said program product is run on a computer or network device.

13. Computer data signal embodied in a carrier wave and representing a program that

instructs a computer to perform the steps of the method of anyone of claims 1 to 9.

14. Electronic device for determining and outputting a similarity measure between two data strings each comprising data entities, comprising:
- a component for receiving a first data string of entities and a second data string of entities,
- a processing unit being connected to said receiving component, said processing unit being configured to determine pairs of consecutively following data entities in said first data string, said processing unit being configured to determine the relative positions of said pairs of consecutively following data entities in said first data string, and for allocating a position label to each of said data entities in the first data string, and numbering same data entities according to their relative position in accordance with the position label; said processing unit being configured to determine similar data entities with the same order in said second data string, and to determine the relative positions of said determined data entities in said second data string, , , said processing unit being configured to determine a matching measure by determining how far the relative positions of data entities in said second data string match with the relative positions of consecutively following data entities in said first data string, and said processing unit being configured to output a similarity measure which corresponds to the matching measure of at least one comparison result, and
- an interface being connected to said for processing unit for outputting said similarity measure.

15. Electronic device according to claim 14, further comprising a storage connected to said processing unit for storing received strings and said determined similarity measures.